DOCUMENT RESUME

ED 313 031 IR 014 085

AUTHOR Abreu, Jose Luis; And Others
TITLE Software Tools: EPICUR.
PUB DATE 88
NOTE 9p.
PUB TYPE Computer Programs (101) -- Reports - Descriptive (141)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Authoring Aids (Programing); *Computer Assisted Instruction; *Computer System Design; *Courseware; Elementary Secondary Education; Instructional Design; Programing Languages

ABSTRACT

EPICUR (Integrated Programing Environment for the Development of Educational Software) is a set of programming modules ranging from low level interfaces to high level algorithms aimed at the development of computer-assisted instruction (CAI) applications. The emphasis is on user-friendly interfaces and on multiplying productivity without loss of quality. EPICUR has been designed and developed using modern techniques of software engineering. Version 2.0 of EPICUR is being implemented in JP1 TopSpeed Modula-2 and in Turbo Pascal 5.0, under the MS-DOS operating system. It is noted that many of EPICUR's tools are general enough to be used by a wide range of programmers. This paper describes the features of EPICUR and provides two examples to illustrate the procedures for producing basic graphics and text. The paper concludes with a description of ELECTRA, an application of EPICUR in which the basic idea is to establish relations between objects. (Author/GL)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Software Tools: EPICUR (*)

Jose Luis Abreu
Albert Fabrega
Marta Olivero

Programa d'Informatica Educativa
Departament d'Ensenyament
Generalitat de Catalunya
Avda. Francesc Cambo, 21, 4at.
08003 - BARCELONA (SPAIN)

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED313031

Abstract: EPICUR (Integrated Programming Environment for the Development of Educational Software) is a set of programming modules ranging from low level interfaces to high level algorithms aimed at the development of CAI applications. The emphasis is on friendly user interfaces and on multiplying productivity without loss of quality. EPICUR has been designed and developed using the modern techniques of software engineering. Version 2.0 of EPICUR is being implemented in JP1 TopSpeed Modula-2 and in Turbo Pascal 5.0, under the MSDOS operating system.

1. - Introduction.

The field of educational software is having a spectacular growth. More and more people are becoming involved in the development and application of CAI. This is probably due in part to the existence of modern author systems and language compilers offering some powerful programming tools.

At first glance two levels of authors of CAI materials can be distinguished. There are the teachers that write "Little programs" to assist them in the presentation of some classroom topics. They are no professional programmers and are willing to use any tool that will reduce their programming effort. With time some become proficient in the use of their programming tools and the quality of their products increases deserving wider distribution. These form a second level of CAI authors.

Materials intended for use only by one particular teacher and in one particular classroom may not require high quality and reliability, but the fact is that many times these products contain valuable ideas that could be profitably applied by other teachers with other students, were the materials suitably developed.

(*) This is a draft of a paper to be submitted for publication elsewhere.

IR014085

PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Albert Fabrega
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Unfortunately in most communities only a handful of teachers have the time and resources to put in the field more than their enthusiasm and a few working hours. The results of the efforts of educational software developers can be significantly improved if their programming tools are improved. And it is clear that they can be improved. Most programming tools are presented as long lists of procedures poorly implemented, poorly documented and with a total lack of structure. Products developed with such tools often show the same weaknesses as the tools with which they are developed.

Certainly, one is always grateful to get a good powerful system of simple tools that will help in writing programs and such systems must be used. But simple things need not be unorganized or unreliable.

The development of EPICUR is motivated by this idea: to offer a simple, reliable and powerful programming environment to the educational software developer. It is not conceived as an exhaustive set of universal programming tools, but we believe that many of EPICUR's tools are general enough to be useful to a wide range of programmers. Our aim is to provide as many teachers as possible with a reasonable set of services that are suitable for solving their programming problems in CAI, and give them the necessary high quality standard that educational products require and deserve.

2. - What EPICUR is.

EPICUR consists of:

(1) A set of data types which are useful in educational software (graphic windows, icons, text fonts, images...).

(2) A library of procedures that manage the previous data types. The procedures are structured in modules. This library is the main component of EPICUR. There are five kinds of modules in it:

(i) Communications modules. These are modules for handling most peripheral equipment (disks, keyboard, printer, mouse, videodisc, networks,...).

(ii) Low level modules for the direct manipulation of the predefined data types (storage and display).

(iii) Intermediate level modules for the orderly management of the structured variables (window processes, menus,...).

(iv) High level modules for the development of educational software. These modules assist the programmer in developing educational software with little effort, without worrying about technical aspects and with the assurance that the end product will be reliable.

(v) Application modules on specific curricular topics (mathematics, languages, ...). These modules combine the procedures included in the general modules with the algorithms needed for handling the special data types of the subject.

(3) Author systems that facilitate the use the simplest parts of EPICUR and result in high degree of productivity in the development of tutorial software.

(4) A set of utilities developed to help create and edit the data types of EPICUR: font editor, icon editor, converters from several data types to the data types of EPICUR and interfaces to transfer the corresponding information.

(5) Exhibition programs for "Demos" that illustrate the possibilities of the environment and give specific examples of their use.

(6) Open application programs created under EPICUR's environment. These are tools with which teachers can generate educational materials with the computer and present such materials to the students in the form of interactive activities. Open application programs consist of two parts:

(i) The first one present the topic to the student. This is done in the form of an interactive activity with the computer through which the student will learn the topic.

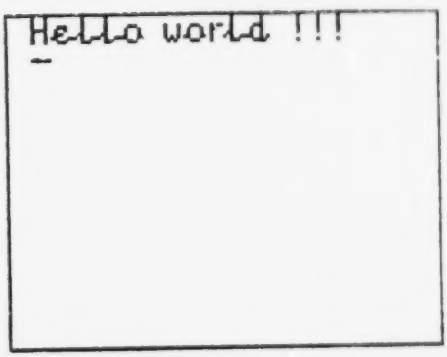
(ii) The second one is a tool with which the teacher can prepare certain materials to be presented to the student by means of the part described in (i).

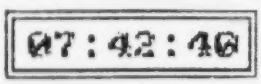
3. - Two examples.

There are some modules in EPICUR designed to communicate a program with the keyboard and the screen in the PC's family of graphic displays and t help in the treatment of text and graphs. The following are two xamples that show some of the procedures in these modules. Figure 1 shows a more sophisticated screen where basic graphics, keyboard and text modules are used, and where four concurrent processes are taking place. All four processes use procedures from EPICUR.

```
MODULE EXAMPLE 1;

FROM INPUT IMPORT INPUT_ReadChar;
FROM KBD IMPORT KBD_OpenForEco,KBD_EcoMode,KBD_SetEcoMode,
                KBD_CloseForEco,KBD_Device,KBD_ErrorInEco;
FROM TXT IMPORT FontType, TXT_LoadFont,TXT_SetFont,
                TXT_SetColor,TXT_WriteStr;
FROM SCR IMPORT SCR_InitMode;

VAR ch:CHAR;
    fnt:FontType;


BEGIN
  SCR_InitMode (5);                      (*CGA color 320*200 *)
  TXT_LoadFont (Fnt, 'font2.car');   (*load font from disk *)
  TXT_SetFont (fnt);                     (*set font for use *)
  TXT_SetColor (1);                      (*set color for text *)
  KBD_OpenForEco('kdb.eco'); (* open file for input echo *)
  KBD_SetEcoMode(KBD_EcoMode(Screen,Disk,Printer))'
                (* input echo to screen, disk and printer *)

REPEAT
     INPUT_ReadChar (ch); (* character read is echoed to
                             screen, disk and printer.  Echo on
                             screen using font 'font2.car'
                             Echo on disk to file 'kdb.eco'*)
UNTIL ch=CHR(13);
KBD_CloseFonEco;     (*close echo file *)
IF KBD_ErrorInEco()  THEN  (*test for succesful echo *)
 TXT_WriteStr('Error in echo device')
END
END EXAMPLE 1.
```

```
MODULE EXAMPLE 2;

FROM SCR IMPORT PIXELMASK;
FROM GR  IMPORT GR_SetViewPort,GR_ClearViewPort,
                GR_ViewPortFrame,GR_SetModeNumber,
                GR_SetDrawMode;
FROM CONICS IMPORT EJES,  SetScale, ORIGEN, CONICA;
FROM Lib IMPORT Delay;
FROM KBD IMPORT KBD_KeyPressed;
CONST Pi = 3. 141592;
VAR t:  LONGREAL;
BEGIN
 GR_SetModeNumber (5);          (*CGA Color 320*200 *)
 GR_SetViewPort (8,8,159,99); (*set viewport for display *)
 GR_ClearViewPort (0);          (*clear viewport *)
 SetScale (6.0);                 (*fixes viewport scale *)
 ORIGEN(-1.0,-1.0);             (sets location of origin *)
 GR_ViewPortFrame (1);          (*draws frame around
                                     viewport *)
 EJES (2);                        (*draws coordinate axis *)
 t:=0.0;
 GR_SetDrawMode (XorMask);     (* xor plot *)
 REPEAT
      CONICA (1.0,0.4,0.6,0.8,t,3); (*draw ellipse *)
      Delay (60);
      CONICA (1.0,0.4,0.6,0.8,t,3); (*erase ellipse *)
      t:=t+Pi/12.0;
 UNTIL KBD_KeyPressed ();
 GR_SetDrawMode (NoMask);            (*normal plot *)
 GR_SetModeNumber (3);          (*text mode *)
END EXAMPLE 2
```





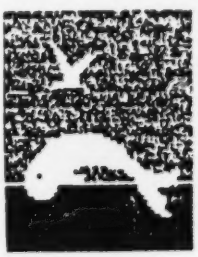


figure 1

4. - ELECTRA: An application of EPICUR.

ELECTRA is an open application program based on EPICUR. It corresponds to the kind of programs explained in 2 (6). In fact ELECTRA is an environment to create a particular class of CAI programs where the basic idea is to establish relations between objects. From the conceptual point of view the student must find relations between the objects of two sets. The relations follow a certain criterium. So the student must do some kind of activity around a relation between the objects of two sets. This activity can take one of a variety of forms consistent with one of the modes of operation of ELECTRA. In all cases the relation between the objects is one to one.

Basically there are three kinds of activities in ELECTRA: Association, Ordering and Virtual Keyboard activities. The first group is illustrated in Figure 2. The student must find for each object from the first set the corresponding object in the second set. These activities can take the reverse form, where the objects are initially hidden. These are "memory games".

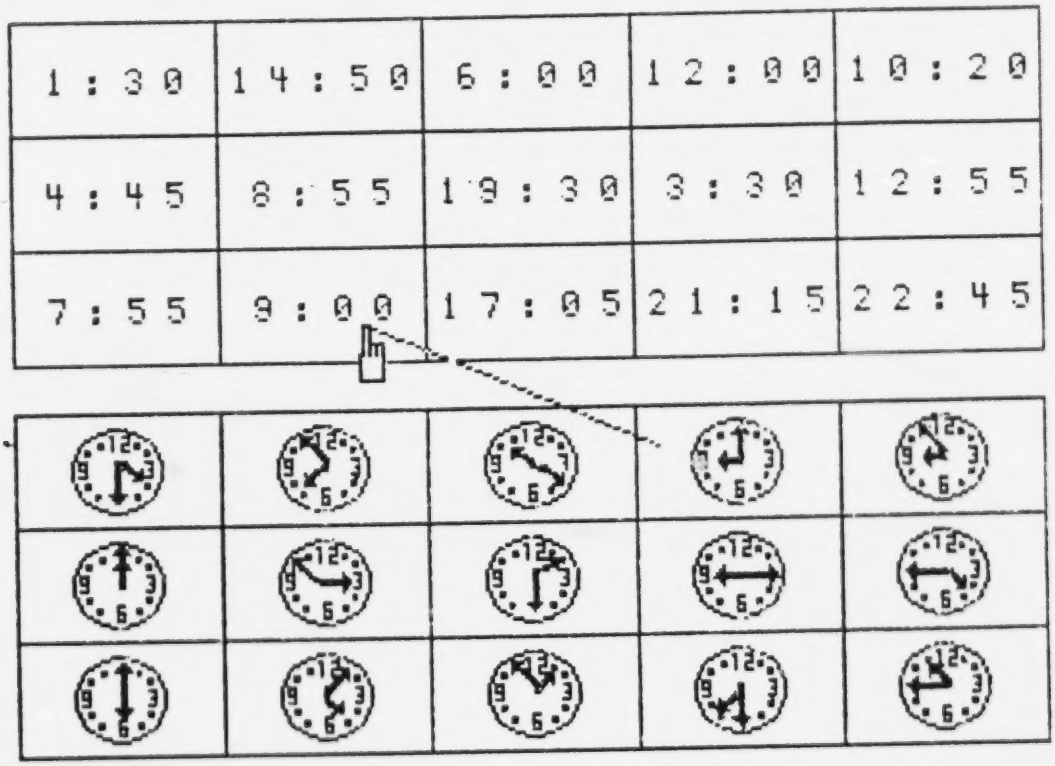


Figure 2

The second group is illustrated in figure 3. The student must find the correct place for each one of the objects in the first set in the template corresponding to the second set.

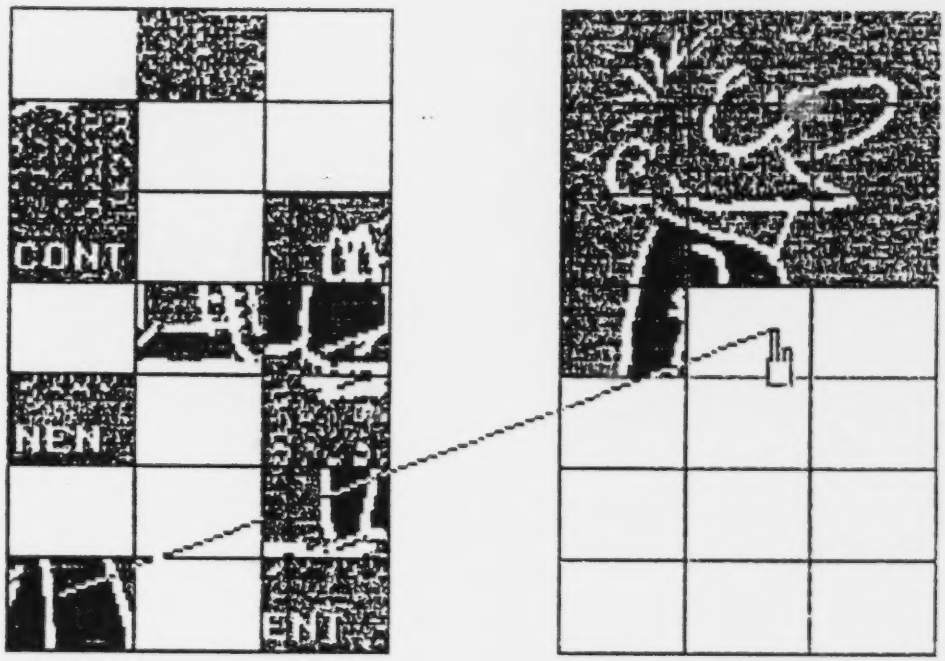


Figure 3

The third group is illustrated in figure 4. In this kind of activities the student either "hits a key" in order to get information about some objects or must "hit the correct Key" in order to answer some question presented in the virtual screen.

Have you seen who lives in the

apple? It is a worm.

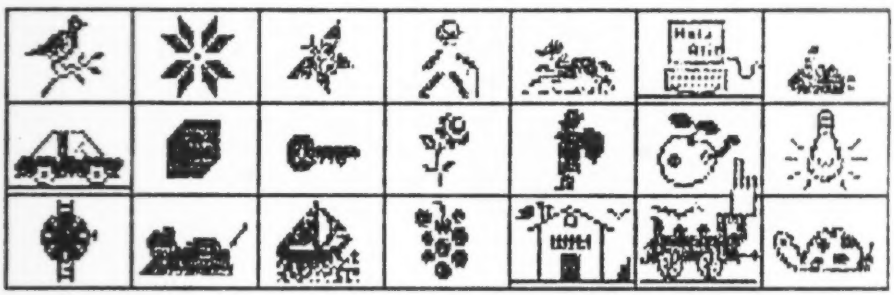


Figure 4

For each form of activity there are different possible distributions on the screen. These are called the geometries of ELECTRA.

With the program comes a set of utilities with which the teacher can make templates, activities and sessions according to his personal taste and needs.